KOPI

(54) Title: METHOD OF PERFORMING MOTION ESTIMATION

	MV2	MV3
MV1	MV	
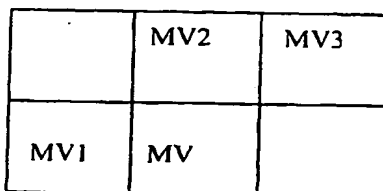
MV : Current motion vector
MV1: Previous motion vector
MV2: Above motion vector
MV3: Above right motion vector

Application of an embodiment of motion vector estimation/prediction in accordance with the invention.

(57) Abstract: Briefly, in accordance with one embodiment of the invention, a method of performing motion estimation for video coding includes the following. Median motion vector components are determined from a set of neighboring macroblocks that include motion vectors. A window of a predetermined size around a pixel location associated with applying the determined median motion vector components is searched to locate a pixel location that produces the closest match. Many other embodiments in accordance with the invention other than this particular embodiment are also disclosed.

WO 01/99437 A2

METHOD OF PERFORMING MOTION ESTIMATION

<u>BACKGROUND</u>

5

The present disclosure is related to performing motion estimation, and, more particularly, to performing motion estimation for video coding, such as, for example, low-bit rate video coding.

10   As is well-known, motion estimation is a common aspect or component of video encoding. There are a variety of techniques of performing video coding, some of which are related to standards and some of which are not. Popular standards include: ITU-T "Video coding for low bit-rate communications," ITU-T Recommendation H.263, version 1, Nov. 1995 and version 2, Jan. 1998; "Generic Coding of Moving Pictures and Associated Audio Information: Video," ISO/IEC 13818-2: International Standard 1995; and "Coding of audio-visual Objects-Part 2: Visual Amendment 1; Visual extensions," ISO/IEC 14496-2: Draft of January 6, 2000; respectively, referred to as H.263, H.263+, MEPG-2, and MPEG-4, hereinafter. These are examples of decoding/decompression standards. Typically, these standards or specifications for these standards provide little or no guidance about compression methodologies that may be employed, including in connection with motion estimation.

Motion estimation is employed to remove temporal redundancies in video frames so that the bandwidth of the communications channel used to transmit the frames may be utilized more efficiently. Motion estimation is typically accomplished by employing block search techniques. A 'full motion' search methodology, in which essentially all pixel locations are searched within a search window, is typically computationally very complex

and, at times, may be prohibitive for real-time applications. In the case of full search

motion estimation, where macroblocks are employed, the number of search points is

1,024 for a 32 x 32 window.

5   To reduce this computational complexity, different approaches have been

proposed. A popular approach is based on or referred to as a logarithmic search. In this

approach, instead of searching every search point or pixel location within a search

window, nine initial points, which are separated by a quarter of a search window, are

checked or searched. After finding a search point or pixel location from these nine which

10   provides the least sum of absolute difference (SAD) value or the least value from the nine

of another error measurement, like mean-squared error (MSE), the search is continued at

eight additional points that are centered about that point by reducing the distance

between search points by half. This continues until the distance between two search

points is one pixel apart. In this approach, 33 search points produce a result, in

15   comparison with 1,024 search points for a full motion search. It is noted that each

additional search point adds additional computation.

Although a logarithmic search reduces the number of search points, this approach

has some disadvantages. For example, the quality of the image produced from this

20   motion estimation approach is generally degraded relative to a full motion search.

Furthermore, with 33 search points, this approach, in some instances, still may not be

suitable for real-time applications. Therefore, a need exists for an approach or technique

that is comparable, or even better, in terms of image quality and yet less computationally

complex than the logarithmic search approach.

25

BRIEF DESCRIPTION OF THE DRAWINGS

The subject matter regarded as the invention is particularly pointed out and

5   distinctly claimed in the concluding portion of the specification. The invention, however,

both as to organization and method of operation, together with objects, features, and

advantages thereof, may best be understood by reference to the following detailed

description when read with the accompanying drawings in which:

10   FIG. 1 is a schematic diagram illustrating motion vectors associated with

macroblocks for use in an embodiment of a technique for motion vector prediction in

accordance with the present invention;

FIG. 2 is a table illustrating simulation results of applying an embodiment in

15   accordance with the invention to motion estimation;

FIG. 3 is a schematic diagram illustrating an embodiment of integer pixel motion

estimation;

FIG. 4 is a plot illustrating the application of an embodiment in accordance with

20   the present invention to an example involving motion vectors; and

FIG. 5 is a table illustrating simulation results of applying another embodiment in

accordance with the invention to motion estimation.

25   DETAILED DESCRIPTION

In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the invention. However, it will be understood by those skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components

5   and circuits have not been described in detail so as not to obscure the present invention.

As previously indicated, motion estimation is employed in video encoding. MPEG-2 and MPEG-4 and other video standards are examples where it is applied for video encoding. However, these methodologies typically do not specify the compression

10   techniques to employ, particularly in connection with motion estimation. As indicated, motion estimation is typically employed to remove temporal redundancies when transmitting video frames in order to make a more efficient use of a limited amount of bandwidth available to transmit the video frames from one location to another, for example. Of course, video coding and decoding is also used in other situations, such as

15   in storage, and the invention is not limited to a particular application of these techniques.

One technique of motion estimation is referred to as full motion search. However, this approach is computationally complex and typically prohibitive for real-time applications. In the case of full search motion estimation, the number of search points in

20   a 32 x 32 window, which would be employed in connection with a macroblock, is 1024. The term macroblock is well-known and well-understood and will not be explained in any great detail; however, whereas it typically refers to a portion of a video frame comprising a square array of luminance pixels, the array having dimensions 16 x 16, and two corresponding square arrays of chroma pixels, each array having dimensions 8 x 8, in

25   this context, it refers to the luminance array and not the chroma arrays. This is the convention applied for this particular embodiment, because, typically, the motion vectors

4

for chrominance are derived from those for the luminance array; however, of course, the invention is not limited in scope in this respect.

To reduce computational complexity, several techniques have been proposed. As previously indicated, a popular approach is based on a logarithmic search. In this approach, as previously described, instead of searching all of the search points within a search window, initial points are searched which are apart from each other by a quarter of the search window size. After finding a point or pixel location of the nine points which gives the least sum of absolute difference (SAD) value or some other measure, the approach then considers or checks eight additional points that are centered about that point by reducing the distance between search points by half. This approach continues until the distance between two search points becomes one pixel apart. As a result, 33 search points are checked in comparison with 1,024 for a full motion search. It is noted that here only the luminosity pixel signal data is employed. As previously indicated, although this approach reduces the complexity in comparison with the full motion search, it still may not be suitable for some real time applications, and furthermore, may produce lower quality video images or frames.

In contrast, an embodiment of a method of performing motion estimation in accordance with the present invention reduces computational complexity and enhances compression performance both in terms of the quality of the image produced and the compression efficiency that results, when compared, for example, with the logarithmic search method. In this particular embodiment, on the transmitting side of a communications channel, for example, a method of performing motion estimation includes the following. Median motion vector components are determined from a set of neighboring macroblocks that include motion vectors. It is noted, of course, that while this particular embodiment employs macroblocks, the invention is not limited in scope in

this respect. For example, other portions of a frame other than a macroblock may be employed. Macroblocks are employed in this embodiment, however, because macroblocks are well-known and, thus, provide convenience, but the invention is in no way restricted to a particular number of pixels or a particular shape or arrangement of

5   pixels to which an embodiment in accordance with the present invention is to be applied.

Next, a window of a predetermined size and shape around a pixel location associated with the determined median motion vector components is searched. More particularly, the median motion vector components are applied to the macroblock to

10   produce a pixel location and that pixel location is the center of the search window, for this particular embodiment. In addition to searching a window of a predetermined size and shape, a pixel location associated with a motion vector having a zero value for all of its components is also search or checked. Therefore, the pixel locations of the window and a pixel location associated with a zero motion vector are checked or searched to

15   determine, of these, which pixel locations produce the closest match with the particular macroblock to which this technique is being applied.

Of course, as has been previously indicated, the invention is not limited in scope to a particular embodiment and many variations are possible. For example, the zero

20   value motion vector may be omitted from the search in an alternative embodiment. Likewise, the closest match when searching pixel locations may be determined by applying the sum of absolute differences (SAD) between the particular macroblock and a macroblock centered at the pixel location being checked or searched, although the invention is not limited in scope in this respect. However, alternatively, the least mean

25   square error (MSE) approach may be employed, or any one of a number of other possible approaches to measure the error or differences may be employed. Likewise, the median motion vector components in this particular embodiment are determined from a set of

neighboring macroblocks, and, as illustrated in FIG. 1, in this embodiment, specifically

three. Likewise, again, the invention is not limited in scope to any particular approach

regarding neighboring macroblocks and any one of a number of approaches

characterizing the set of neighboring macroblocks employed are possible. For example,

5   fewer macroblocks or more macroblocks may be employed. Likewise, in alternative

embodiments, successive or sequential macroblocks may not necessarily be employed.

     Furthermore, in this particular embodiment, although the invention is again not

limited in scope in this respect, to determine the closest match, as previously indicated,

10   luminosity pixel signals values are employed. However, where luminosity pixel signals

values are not available, as one alternative example, the component of the frame that

contains the greatest amount of luminosity signal information may be employed.

Likewise, luminosity may alternatively not be employed and, instead, chroma signal

values or signal values where the chroma components are dominant may be employed.

15   Again, the invention is not limited in scope to a specific approach. For example, where

frames are provided in Red-Green-Blue (RGB) color space format, typically, the green

pixel signal values will be employed, although, again, the invention is not limited scope in

this respect.

20   Although the invention is not limited in scope in this respect, for MPEG-4 and

H.263, when interframe coding is employed, the motion vector is coded and transmitted.

A compressed video data stream, as transmitted, typically comprises codes for motion

vectors generated by motion estimation as well as codes for error frames. In such a

situation, and in this particular embodiment, a motion vector, MV, may be represented by

25   two components, (MV_X, MV_Y), where MV_X and MV_Y are, respectively, the components in

the horizontal and vertical directions. Typically, a motion vector may be calculated as

follows, referring to FIG. 3:

$$SAD = \min_{(x,y) \in S} \sum_{j=0}^{15} \sum_{i=0}^{15} \left| C[i,j] - R[x_0 + x + i, y_0 + y + j] \right|$$

where

(x_0, y_0)	Upper left corner coordinates of the current macroblock
C[x,y]	Current macroblock luminance samples
R[x,y]	Reconstructed previous frame luminance samples
S	Search range: $\{(x,y) : -16 \le x, y < 16\}$
(MVx, MVy)	motion vector that results in the lowest sum above

In this particular embodiment, to reduce the number of bits employed for motion vector coding, motion vector components, horizontal and vertical in this embodiment, are coded differentially by using a spatial neighborhood of three macroblocks, each of the macroblocks having a motion vector, as illustrated in FIG. 1. It is also noted that this neighborhood signal information has already been transmitted in this embodiment. In this embodiment then, these three motion vectors are candidate predictors for differential coding of the motion vector to currently be coded. It will, of course, be appreciated that, as indicated previously, the invention is not restricted in scope to employing the three previous macroblocks or, more particularly, the motion vectors from the three previous macroblocks. Any one of a number of previous macroblocks may be employed and the macroblocks need not be successive, as previously indicated for this particular embodiment.

In this embodiment, the motion vector coding is performed independently for the horizontal and vertical components. For each component, in this embodiment, the median value of the three candidates for a component is computed as follows:

$$Px = Median(MV1x, MV2x, MV3x)$$
$$Py = Median(MV1y, MV2y, MV3y)$$

5

For example, if MV1 equals (-2,3), MV2 equals (2,5), and MV3 equals (-1,8), then the median motion vector components, designated P_X and P_Y here, are -1 and 5 respectively. As indicated by the following equations:

$$MVDx = MVx - Px$$
$$MVDy = MVy - Py$$

10

An aspect of this approach is the observation that, typically, spatial correlation is present for the motion vector field among neighboring macroblocks. Therefore, an efficient approach to searching may include using a smaller search window centered at

15   P_X, P_Y. Furthermore, here, by calculating the median values separately for each vector component, the opportunity for additional information is present. In particular, if the median motion vector components come from the same macroblock, this may indicate that the motion vector field is relatively 'smooth,' meaning, in this context, that there is relatively little variation between motion vectors that are located in relatively close spatial

20   proximity. Therefore, for this embodiment, the search window is limited to nine points centered around P_X, P_Y. However, alternatively, if the median motion vector components, here horizontal and vertical, respectively, come from different macroblocks, this may indicate that the motion field is more complicated and, therefore, the search window is increased to 25 points, although, again, centered at P_X, P_Y. Furthermore, in this

25   embodiment, a zero motion vector for both components is also searched, although, as

9

previously indicated, the invention is not limited in scope in this respect. A reason for employing a zero motion vector is because where it provides an acceptable match in terms of error, such as SAD, it conserves bandwidth, which in some situations may be particularly desirable. As previously indicated, in this particular embodiment, a sum of

5   absolute differences (SAD) is employed, although the invention is not limited in scope in this respect.

Using pseudocode, this particular technique or embodiment in accordance with the invention may be characterized as follows:

10   If (Px=MV1x) then i=1
            else if Px=MV2x then i=2
            else i=3

            If (Py=MV1y) then j=1
15   else if (Py=MV2y) then j=2
            else j=3

            If (i==j) then
                    Apply motion estimation using 3x3 search window centered by
20   (Px, Py);
            else
                    Apply motion estimation using 5x5 search window centered by
            (Px, Py);
            endif;

25

            FIG. 2 is a table comparing efficiency for this particular embodiment when compared with a full motion search and a logarithmic search. In a full search, the search

range is (-16, 15) so that the number of search points is 1,024, as previously indicated. Likewise, as previously indicated, the logarithmic search includes 33 search points. The number of search points for this particular embodiment, of course, depends in part on the video sequence or sequences. In this particular set of sequences to which this

5 embodiment was applied, a smaller search window was applied 69% of the time so that the number of search points is a weighted average of 26 and 10, providing, for this example, 15 search points. These results were obtained by using image sequences with Quarter Common Intermediate Format (QCIF) sized images for 134 frames. The frame rate applied here was 10 frames per second.

10

The results in the table of FIG. 2 indicate less complexity where motion estimation is performed for this embodiment in accordance with the invention. This is a desirable result since, typically, motion estimation comprises a significant portion of a video encoder computational burden. The number of search points for this particular

15 embodiment is only 1.5% of the number of search points employed for a full search, a significant reduction in computational burden. Likewise, the bit rate is reduced by more than 4.5% for this particular embodiment, when compared with the full search, although the degradation in signal-to-noise ratio is only about .45 to .65 decibels. Likewise, as the table illustrates, the peak signal-to-noise ratio (PSNR) value obtained for this particular

20 embodiment is better than that obtained for the logarithmic search with less complexity and a smaller "bit budget."

As the previous discussion illustrates, this embodiment in accordance with the present invention has a number of advantages over state of the art approaches to motion

25 estimation and encoding. As previously indicated, these advantages include a reduction in computational burden, an improved compression ratio, a reduction in bits for motion vector coding, and an improved peak signal-to-noise ratio relative to a logarithmic search.

11·

Furthermore, although the invention is not limited in scope in this respect, this particular embodiment is compliant with both MPEG-4 and H.263 for very low bit rate coding. Likewise, this embodiment may be implemented in hardware, software, firmware, or any combination thereof, as desired.

5

Another embodiment in accordance with the invention is illustrated by the following pseudo-code, where Rx and Ry are calculated as follows:

Rx=MAX(MV1x,MV2x,MV3x)-MIN(MV1x,MV2x,MV3x)

10   Ry=MAX(MV1y,MV2y,MV3y)-MIN(MV1y,MV2y,MV3y)

If (Px=MV1x) then i=1

else if Px=MV2x then i=2

15   *else i=3*

If (Py=MV1y) then j=1

else if (Py=MV2y) then j=2

else j=3

20

If (i==j) {

Apply motion estimation using x_arm x y_arm search window centered by (Px, Py) where

x_arm = 3 and y_arm = 3;

}

25   *(*

else {

     . set x_arm = 5 and y_arm = 5;

     If (Rx<=3) set x_arm = 3;

if (Ry<=3) set y_arm = 3;

}

5   This particular embodiment provides an advantage of less computational complexity or burden than the previous embodiment, although, there may be a sacrifice in terms of image quality as a result. This embodiment is similar to the previously described embodiment in that the median values are calculated and used to determine or deduce properties about the motion vector field. As previously, if the median values for both

10   coordinates come from the same macroblock, in this embodiment, the search is limited to 9 points. However, if the median values for the coordinates come from different macroblocks, then the number of points to search may or may not be increased. Rather than simply searching a 25 point window in this circumstance, as the in the previous embodiment, instead, the range in the different directions are calculated and that

15   information is employed to make decisions about the search window.

The range values, Rx and Ry, provide an indication about how the motion vector field may be changing in these directions. Therefore, is the change is relatively large, then a larger window in the direction is searched; however, if the change is relatively

20   small, then a smaller window in that direction is searched. This reduces complexity in comparison with the previous embodiment, because in some cases, a 9 point or 15 point search will be employed for the latter embodiment, whereas a 25 point search would have been employed for the former embodiment.

25   Application of this particular embodiment is specifically illustrated by an example in FIG. 4. The motion vectors are (4,2), (9,3), and (5,4). Therefore, a 3 x 5 rectangle centered at (5,3) is searched if the previously described embodiment is applied.

Likewise, FIG. 5 is a table illustrated the results of applying this embodiment to the

sequences for which the previous embodiment was evaluated. FIG. 5 provides a

comparison of results for this embodiment with results for a full search, logarithmic

5   search, and the previous embodiment.

It will, of course, be appreciated that the invention is not restricted in scope to a

particular embodiment or implementation, as previously indicated. For example, the

foregoing approach, as one example of an approach in accordance with the invention,

10   may be implemented in hardware, in software, in firmware, and/or any combination

thereof. Again, intended merely as examples that do not limit the scope of the invention,

an embodiment may comprise an imager including hardware, such as integrated circuit

chips, that implement the foregoing. Alternatively, the imager may be coupled to a

computing platform that includes software capable of implementing the foregoing.

15   Likewise, a digital camera coupled to a desktop personal computer, for example, may

implement an embodiment. Furthermore, these implementations in hardware and

software may, of course, deviate from the foregoing and still be within the scope of the

present invention.

20   For embodiments that are at least partially implemented in software, such as, for

example, the previously described embodiment, such software may reside on a storage

medium, such as, for example, random access memory, a CD ROM, a floppy disk, or a

hard drive, such that instructions are stored, which, when executed, such as by a

computing platform, such as a PC or other computing device, so that the system is

25   capable of executing the instructions to result in motion estimation. Likewise, such

software may reside in firmware also, such as in flash memory or EEPROM, for example.

While certain features of the invention have been illustrated and described herein, many modifications, substitutions, changes and equivalents will now occur to those skilled in the art. It is, therefore, to be understood that the appended claims are intended to cover all such modifications and changes as fall within the true spirit of the invention.

5

<u>Claims:</u>

1.   A method of performing motion estimation for video coding comprising:

     determining median motion vector components from a set of neighboring

5   macroblocks that include motion vectors; and

     searching a window of a predetermined size around a pixel location associated

with applying the determined median motion vector components to locate a pixel location

that produces the closest match.

2.   The method of claim 1, wherein, searching includes searching a pixel location

10   associated with a motion vector having zero valued components.

3.   The method of claim 2, wherein the motion vector estimation is applied to a

predetermined portion of a video frame.

4.   The method of claim 3, wherein the predetermined portion of a video frame

comprises a particular macroblock;

15   and further comprising:

     coding a motion vector for the particular macroblock based, at least in part, on the

difference between the median motion vector components and the pixel location that

produces the closest match from the potential pixel locations.

5.   The method of claim 2, wherein the median motion vector components comprise a

20   horizontal motion vector component and a vertical motion vector component.

6.   The method of claim 2, wherein the set of neighboring macroblocks comprises

three previously coded macroblocks.

7.   The method of claim 6, wherein the three previously coded macroblocks comprise

three macroblocks coded immediately previously.

8.   The method of claim 2, wherein the pixel location that produces the closest match

is determined by applying the sum of absolute differences to each pixel location of the

5   potential pixel locations.

9.   The method of claim 8, wherein the closest match comprises the closest match of

luminance pixel signal values.

10.   The method of claim 2, wherein the pixel location that produces the closest match

is determined by applying the mean square error (MSE) to each pixel location of the

10   potential pixel locations.

11.   The method of claim 1, wherein the median motion vector components are each

determined from the set of neighboring macroblocks independent of the other

components.

12.   The method of claim 11, wherein if the median motion vector components each

15   come from different macroblocks in the set of neighboring macroblocks, then the window

comprises a five pixel by five pixel window.

13.   The method of claim 11, wherein if the median motion vector components each

come from different macroblocks in the set of neighboring macroblocks, then the range of

the motion vector components are computed.

20

14.   The method of claim 13, wherein if the range of a particular component is equal to or below a predetermined value then the window searched is three pixels long in that particular component direction, and five pixels long otherwise.

15.   The method of claim 14, wherein the predetermined value is three pixels.

5

5   16.   The method of claim 13, wherein if the range of a particular component is below a predetermined value then the window searched is a predetermined integer value X in that particular component direction, and a predetermined integer value Y otherwise, where Y is greater than X.

17.   The method of claim 11, wherein if the median motion vector components each

10   come from the same macroblock in the set of neighboring macroblock, then the window comprises a three pixel by three pixel window.

18.   An article comprising: a storage medium, having stored thereon, instructions capable of being executed by a computing platform, said instructions when executed by said platform, resulting in:

15   determining median motion vector components from a set of neighboring macroblocks that include motion vectors; and

          searching a window of a predetermined size around a pixel location associated with applying the determined median motion vector components to locate a pixel location that produces the closest match.

20

20   19.   The article of claim 18, wherein, searching includes searching a pixel location associated with a motion vector having zero valued components.

20.   The article of claim 19, wherein the instructions, when executed, result in a motion

vector for a particular macroblock being estimated based, at least in part, on the pixel

location that produces the closed match from the potential pixel locations;

        and further result in:

5   coding the motion vector for the particular macroblock based, at least in part, on

the difference between the median motion vector components and the pixel location that

produces the closest match from the potential pixel locations.

21.   The article of claim 20, wherein the pixel location that produces the closest match

is determined by applying the sum of absolute differences to each pixel location of the

10   potential pixel locations.

22.   The article of claim 21, wherein the closest match comprises the closest match of

luminance pixel signal values.

23.   The article of claim 18, wherein the median motion vector components are each

determined from the set of neighboring macroblocks independent of the other

15   components.

24.   The article of claim 23, wherein if the median motion vector components each

come from different macroblocks in the set of neighboring macroblocks, then the window

comprises a five pixel by five pixel window.

25.   The article of claim 23, wherein if the median motion vector components each

20   come from different macroblocks in the set of neighboring macroblocks, then the range of

the motion vector components are computed.

19

26. The article of claim 25, wherein if the range of a particular component is equal to or below a predetermined value then the window searched is three pixels long in that particular component direction, and five pixels long otherwise.

27. The method of claim 25, wherein if the range of a particular component is below a predetermined value then the window searched is a predetermined integer value X pixels in that particular component direction, and a predetermined integer value Y pixels otherwise, where Y is greater than X.

28. The method of claim 23, wherein if the median motion vector components each come from the same macroblock in the set of neighboring macroblock, then the window comprises a three pixel by three pixel window.

29. A system for performing motion estimation comprising:
a computing platform;
said computing platform being adapted to produce median motion vector components from a set of neighboring macroblocks that include motion vectors and being further adapted to search a window of a predetermined size around a pixel location associated with the median motion vector components to produce a motion estimate.

30. The system of claim 29, wherein said platform is further adapted to search a pixel location associated with a motion vector having zero valued components.

31. The system of claim 30, wherein said platform is further adapted to-coding the motion vector for the particular macroblock based, at least in part, on the difference between the median motion vector components and the estimated motion vector.

Simulation Results

|	Full Search	Logarithmic Search	Median Search
Search points	1024	33	(25+1)(31%)+(9+1)(69%)= 15
Bits/frame	6447	6246	6155
PSNR_Y	32.03	31.14	31.38
PSNR_U	37.20	36.53	36.72
PSNR_V	37.95	37.23	37.50

FIG. 2

FIG. 3: Integer Pixel Motion Estimation

Search window centered around Px, Py

(5,4)

X (5,3)

(9,3)

(4,2)
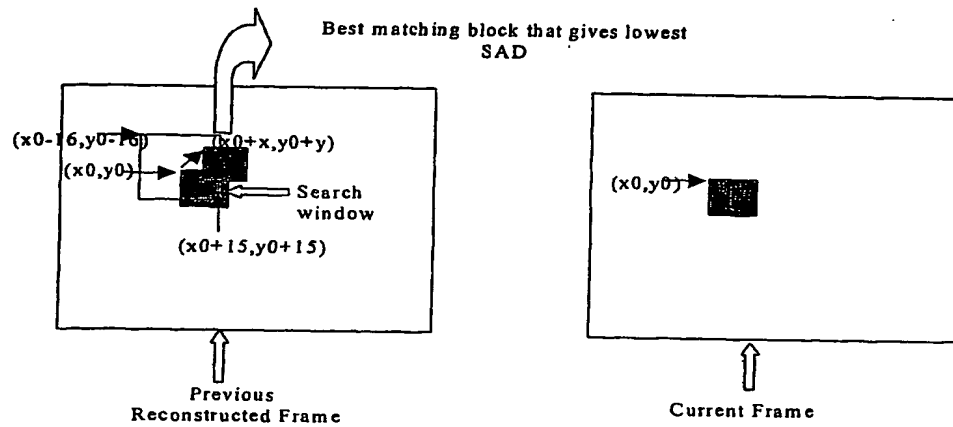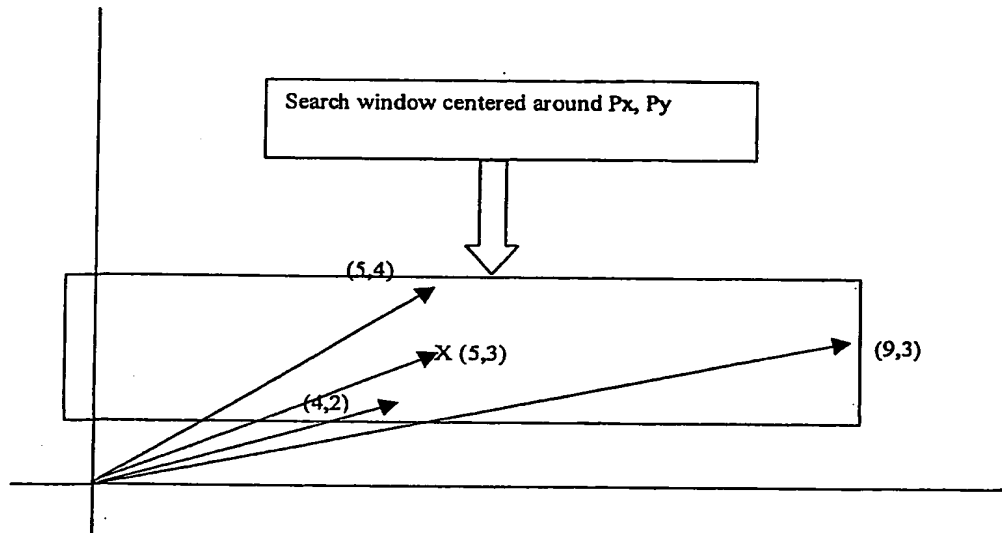
FIG. 4

5/5

Simulation Results

	Full Search	Log. Search	Median Search (6574/9576= 69%)	Alt. Embodiment
Search points	1024	33	26(31%) + 10(69%)= 15	10.69 (see below)
Bits/frame	6438	6475	6456	6435
PSNR_Y	32.15	31.27	31.60	31.57
PSNR_U	37.15	36.65	36.89	36.86
PSNR_V	37.99	37.25	37.62	37.63

Note:
MVx and MVy come from the same block: 6512 (3x3 window)
MVx and MVy come from different blocks: 3064
     5x5: 255
     3x5: 239
     5x3: 175
     3x3: 2395
So it will be approximately
$10 \times (6512+2395)/(6512+3064) + 16 \times (239+176)/(6512+3064) + 26 \times 255/(6512+3064) = 10.69$

FIG. 5